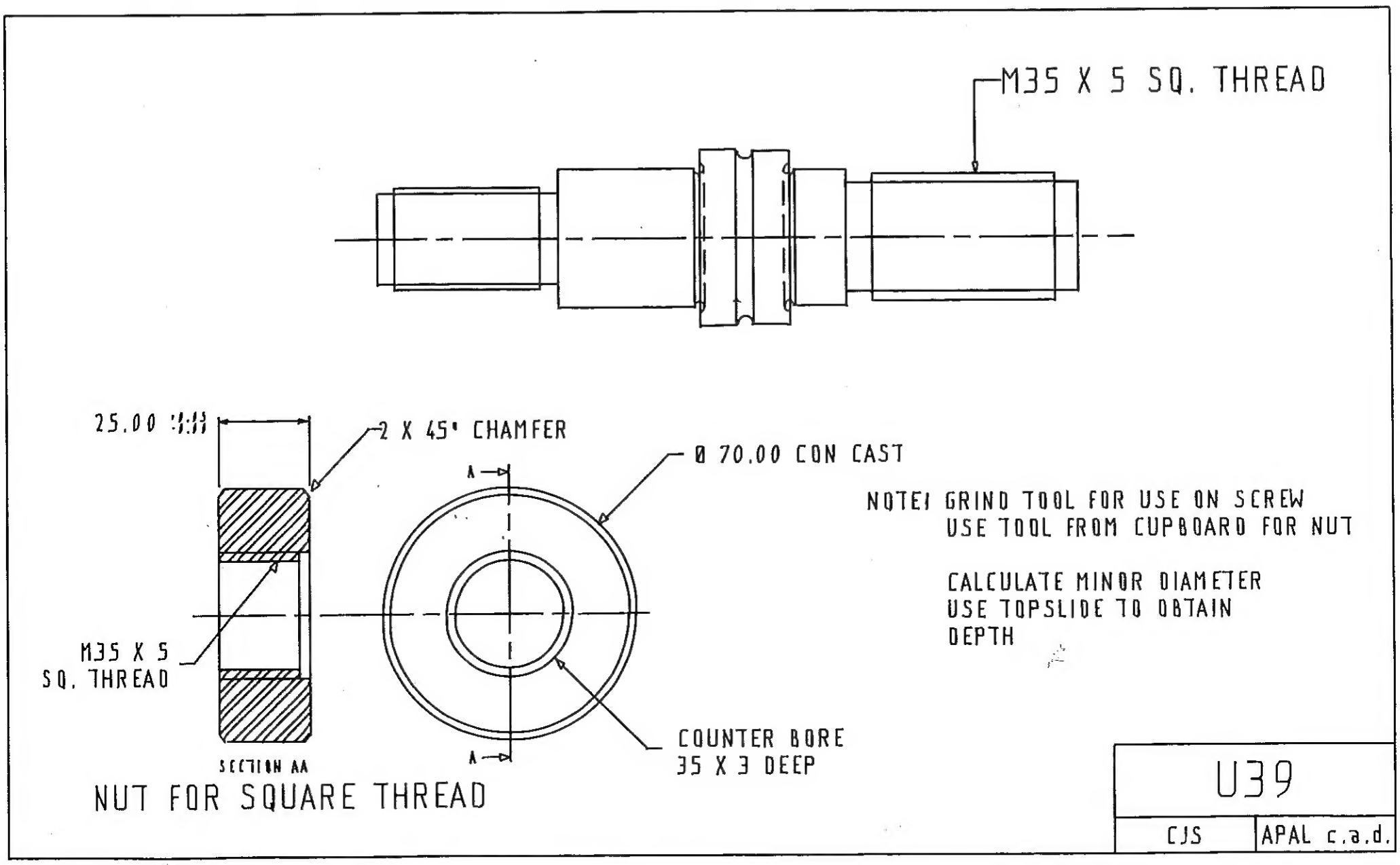
AMIGA
WORKBENCH

$2
RRP

Registered by Australia Post – Publication No. VBG7930

Number 28 | Circulation: 1200 | September 1988



Next AUG Meeting

Sunday, September 18th, 1988 at 2pm

(Doors open at 1pm, meeting starts at 2pm sharp)

AUG meetings are held in the Rotunda at Monash University
Wellington Road, Clayton Melways map 70 reference F10 and map 84A

AUG's AmigaLink BBS - (03) 792 3918

Amiga Users Group Inc, PO Box 48, Boronia, 3155, Victoria, Australia

AMIGA™ Users Group

Who Are We?

The Amiga Users Group is a not-for-profit association of people interested in the Amiga computer and related topics. With over 1000 members, we are the largest independent association of Amiga users in Australia.

Club Meetings

Club meetings are held at 2pm on the third Sunday of each month in the Rotunda at Monash University, Wellington Road, Clayton. Details on how to get there are on the back cover of this newsletter. The dates of upcoming meetings are:

Sunday, September 18th at 2pm
Sunday, October 16th at 2pm
Sunday, November 20th at 2pm

Production Credits

This month's newsletter was edited by Peter Jetson. Equipment and software used was: Non-descript Taiwanese PC Clone computer, Brother HR-40 printer, Brother HL-8 printer, Gemini 10x printer, Wordstar, Fancy Font and Grabbit.

Copyright and Reprint Privileges

Contributions

Articles, papers, letters, drawings and cartoons are actively sought for publication in Amiga Workbench. Please submit your contributions on disk, since that means they don't have to be re-typed! All disks will be returned! Please save your article in **text-only** format (If it can be loaded by ED, it is text-only). Absolute deadline for articles is 16 days before the meeting date. Contributions can be sent to: The Editor, AUG, PO Box 48, Boronia, 3155.

Membership and Subscriptions

Membership of the Amiga Users Group is available for an annual fee of $20. To become a member of AUG, fill in the membership form in this issue (or a photocopy of it), and send it with a cheque for $20 to:

Amiga Users Group, PO Box 48, Boronia, 3155

Public Domain Software

Disks from our public domain library are available on quality 3.5" disks for $8 each including postage on AUG supplied disks, or $2 each on your own disks. The group currently holds over 200 volumes, mostly sourced from the USA, with more on the way each month. Details of latest releases are printed in this newsletter, and a catalog disk is available.

Member's Discounts

The **Amiga Users Group** negotiates discounts for its members on hardware, software and books.

Currently, **Technical Books** in Swanston Street in the city offers **AUG** members a 10% discount on computer related books, as does **McGills** in Elizabeth Street. Just show your membership card. Although we have no formal arrangements with other companies yet, most seem willing to offer a discount to AUG members. It always pays to ask!

Back Issues of Newsletter

All back issues of Amiga Workbench are now available, for $2 each including postage. Note that there may be delays while issues are reprinted. Back Issues are also available at meetings.

AmigaLink - Our Bulletin Board System

The Amiga Users Group operates a bulletin board system devoted to the Amiga, using the Opus message and conferencing software. AmigaLink is available 24 hours a day on (03) 792 3918, and can be accessed at V21 (300bps), V22 (1200bps), V23 (1200/75bps) or V22bis (2400bps) using 8 data bits, 1 stop bit and no parity.

AmigaLink is part of a world-wide network of bulletin boards, and we participate in national and international Amiga conferences. AmigaLink has selected Public Domain software available for downloading, and encourages the uploading of useful public domain programs from its users. AmigaLink is FidoNet node number 3:631/324.

Newsletter Advertising

The Amiga Users Group accepts commercial advertising in Amiga Workbench subject to the availability of space at these rates:

Quarter page	$20
Half page	$40
Full page	$70
Double page spread	$120

These rates are for full-size camera-ready copy **only**. We have no photographic or typesetting facilities. Absolute deadline for copy is 16 days before the meeting date. Send the copy and your cheque to: The Editor, AUG, PO Box 48, Boronia, 3155, Victoria.

Amiga Users Group Committee

Co-ordinator:	Bob Scarfe	376 4143	Kensington
Vice Co-ord:	Lester McClure	233 5664	Mt Waverley
Meeting Chair:	Ron Wail	848 7283	Doncaster
Secretary:	John Elston	375 4142	Moonee Ponds
Treasurer:	Donna Heenan	729 2327	Bayswater
Membership:	Neil Murray	794 5683	Dandenong
Purchasing:	Bohdan Ferens	792 1138	Dandenong
Book Library:	Joan Wood	580 7463	Aspendale
Disk Library:	Craig Hutchison	578 3962	Carnegie
Editor:	Peter Jetson	762 1386	Boronia
Committee:	Doug Myers	428 0682	Abbotsford

DBW Render and I

by Con Kolivas

First I would like to make clear what DBW Render is and then ramble on about my experiences with it. I am the person you would have seen at the front of the July Advanced Graphix meeting giving a demo on this program, so you may read this as additional information, or as an introduction to it if you weren't there.

DBW Render is, as some of you would already know, another RAY tracing program. DBW stands for the name of the person who wrote the original (David B. Wecker), and unlike most programs of this sort, it is Public Domain, but not unlike Public Domain software, it is a fine piece of powerful, useful software, and not quite completed.

For an introduction to it, I will repeat the passage that I read out at the meeting from the program's document guide...

The DBW Render package uses an algorithm called Distributed Ray Tracing. In addition to neatly solving the hidden surface problem, ray tracing (in general) affords the opportunity for doinn excellent (that is highly realistic) shading and illumination calculations.

This package is written in MANX-Aztec C and runs on any AMIGA system. The RAY program knows how to create images composed of four primitive geometric objects; spheres, parallelograms, triangles, and flat circular rings. Some of the features of the program are:

Determination of correct shadows cast by arbitrarily shaped objects, onto arbitrarily shaped objects, due to arbitrarily placed light sources. Diffuse and specular reflections (with arbitrary levels of gloss or polish). Rudimentary modeling of object-to-object diffusely reflected light is also implemented, that among other things accurately simulates color bleed effects from adjacent contrasting colored objects. Mirror reflections, including varying levels of mirror smoothness or perfection. Refraction and translucency (which is akin to variable microscopic smoothness, like the surface of etched glass). Two types of light sources: purely directional (parallel rays from infinity) of constant intensity, and spherical sources (like light bulbs, which cast penumbral shadows as a function of radius and distance) where intensity is determined by the inverse square law. Photographic depth-of-field. That is, the virtual camera may be focused on a particular object in the scene, and the virtual camera's aperture can be manipulated to affect the sharpness of foreground and background objects. Solid texturing. Normally, a particular object (say a sphere) is considered to have constant properties (like color) over the entire surface of the object, often resulting in fake looking objects. Solid texturing is a way to algorithmically change the surface properties of an object (thus the entire surface area is no longer of constant nature) to try and model some real world material. Currently the program has built in rules for modelling wood, marble, bricks, snow covered scenes, water (with arbitrary wave sources), plus more abstract things like color blend functions. Fractals. The program implements what's known as recursive triangle subdivision, which creates all manners of natural looking surface shapes (like broken rock, mountains, etc.). The character of the fractal surface (degree of detail, roughness, etc.) is controlled by parameters fed to the program. AI heuristics to complete computation of a scene within a user specified length of time. When the RAY program runs, it reads a sort of command file, that contains a geometric description of the scene to be rendered, plus desired global lighting conditions, camera parameters, etc. Then the program computes for a long time, writing each scan line to disk as it computes it.

Given that description, you may have a feeling that this program is very similar to Sculpt-3d, and you are not wrong. However, I feel this program is more flexible and powerful than Sculpt. To describe what happens... First you write a simple text file (unlike Sculpt) that contains all the information necessary for the RAY program to generate a picture from it. In the file you include details like the position of the camera lens, the position and various other details of light sources (up to 100 directional and 100 point sources which can have a varying radius as mentioned above), the position of objects in the scene, their texture, color, transparency, reflective properties, index of refraction and so on. The two other commands (optional) that you include are probably what makes this program more flexible than Sculpt. The first describes the time taken for the complete picture to generate - the more time you give it, the more perfect the final product will be - this is unbelievably useful as you would know if you have

used Sculpt. The second command defines what region of the picture you wish to compute, this program having the facility to combine a whole lot of minor generated pictures and form a full screen one. This too is a bonus; not only can you do certain parts at a time overnight, but you can also recover data that you would have lost using Sculpt if the computer crashes - you can't believe how satisfying that is. Now, from the file generated from the RAY program, you run a small secondary program that will generate an IFF file from the first, telling you how many colours (out of 4096 or 32 if you wish) it used.

On to some specifics now. This program, I am led to believe, will work out a picture comparing favourably to Sculpt in one fifth of the time! It runs on an Amiga, without any loss of speed or flexibility, with only 512K! It can even be configured, the text file adds, to run on only 256K! The program does not make spheres out of triangles - they are essentially perfect. The bonus of variable reflectiveness, imperfection, transparency and luminance puts Sculpt almost out of business for "Real world" representations. Add to this the hard-wired optional textures (which are also controlled by the user) and you have one very flexible program!

But, like any powerful program, it has its cons as well as its pros. One of the biggest problems is that it was created for the Public Domain, not as commercial software. Being PD, the program is incomplete, and there are still some bugs in it. It can afford to have bugs in it, as it does contain some 4000 lines of C-code, and is not the simplest of programs. I will not go into detail about all the bugs as the documentation supplied with it should contain my changes to it, describing the problems I have encountered with it. The program was written for NTSC users, meaning it won't create a full screen image, but I have modified this (with my limited knowledge of C) to work in a full sized PAL-screen environment. This, however, takes longer, needing to calculate more scan lines.

As for the types of images I have produced with it... I haven't accomplished that much as I have spent most of my time figuring out how it works and trying out the built in functions of the program and even this has been very impressive, as you would have noticed had you attended the SIG meeting. I have passed on a copy of the DBW Render disk to Dan (one of the many conveners of the SIG) so somebody should be able to copy it for you at the SIG meeting from August onwards. On the disk I have included the two other programs that go well with this program, Zoo and PopCLI for reasons that should be clear to most graphics users.

Since the source code is included, anybody out there with more advanced C programming skills and time than myself can help us (as in the advanced graphix mob) use this program for future purposes. I can't (or couldn't depending on when you read this article) continue my demonstration of this program at the August meeting because of other commitments (like a ski trip), but in the faithful hands of the other members, I am sure the new improved Advanced Graphix SIG will do just fine with other demonstrations and ideas. Thanks for listening.

==========

Big hassles and I

by Con Kolivas

The article above, **DBW Render and I** would have been in the August issue of Workbench but I ran into troubles. The first was PopCLI. I do hope other people have noticed the same problem or won't suffer the same fate. My PopCLI was infected by a particularly nasty time-bomb/trojan horse virus. It would sit in memory for a while, still working like PopCLI should, and then decide it would hang up my Amiga. Like all common Amiga viruses, it would also write itself to the bootblock of any write-enabled disks. I thought I was safe from any form of virus because I had the SCA virus protection program. This new virus, would delete the virus protection from the bootblock and put itself in its place. Because just about all my disks were infected with this virus, I could not get rid of it. Eventually, I managed to disable the validation process using Mirror Hackpack, while at the same time having the SCA virus protection program running. Although the virus protection program did not protect against the new virus, it could remove it from the disk unknowingly, using the protect, or kill virus options. Then I would cold-boot and it was gone. Finally, I went through all my disks that contained PopCLI and deleted it from them. If you have received a copy of DBW Render that contains PopCLI on it, be suspicious unless it contains PopCLI II, a replacement I will provide with the program if people wish a copy of it. The only way to tell them apart is from the message they give on running them.

If virus trouble wasn't bad enough, Amy died. She would come on occasionally and then quit. What would happen was the screen would turn green, the power light would flash eleven times, she would reboot and repeat. This is not disaster. It is a problem that is common in Amiga's with a Fat Agnus chip (as opposed to a plain Agnus in the 1000's). The contacts are not sitting well and old fatso needs to be reseated. If you are under warranty, no problem. If not, you will be charged on labour time only. The process takes about three minutes, so go to the cheapest labour centre you know of. Charlie's in Abbortsford charges a minimum of forty-five dollars an hour! - and that is cheap! I recommend taking it to any place that charges minimum of half hour labour at most (if you can find one, I found one too late out at Croydon in someone's home - MVB Computer Supplies).

Finally, while I still have your attention, I would like to recommend a brilliant new word-processor called Excellence!. It is made by the same people that made Scribble! (Micro-Systems Software), but any resemblance to Scribble! ends about there. It is a full-featured WYSIWYG word-processor with just about everything anyone ever wanted. Mostly, I am recommending it to AUG, since I believe an Amiga users group should at least write their newsletter on an Amiga! It is fitted with the new 1.3 printer drivers, and four new fonts specially designed for laser-printers. It beats the crap out of the old Word-Perfect (the aim of MSS) because it IS friendly, fast and does heaps more. It was designed on an Amiga by Amiga programmers for the Amiga so it utilises the capabilities. I could go on for hours, but all I need to say is that I praise it all the way.

The only thing I don't like is that you have to check the grammar of your document at the end, not while you are typing (which the spelling checker and thesaurus will do). Perhaps it is better that way? This article was written completely on Excellence! but I included nothing fancy so the IBM would be able to handle it. Thanks again for listening.

==========

It Ain't The Right Tool

by Peter Jetson

Okay, I'll have another look at Excellence! When it first came out, I went in a dealer's and spent quite some time looking over the manual, and that only raised more questions than answers. You see, an editor's job is NOT necessarily best served by a word-processor/text editor, because most of those programs assume you are going to enter your text from the keyboard.

Assuming that an editor actually gets contributions (a rather big assumption this month, I'm sorry to say!), the editor just checks spelling, grammar, usage and intelligence, edits out bits that don't make sense, are libelous or whatever, and rearranges whatever he has left into the space he has available. Hardly any of an editor's time is actually spent CREATING text, and unfortunately, that is what most word-processors and text editors are designed for.

Just about every article I get receive has been written using a different program, so at the moment, I have to manually go through each article and reformat it, removing any "non-text" characters as I go. I have designed a program that tries to turn almost anything into just text, but it makes so many mistakes that editing by hand is usually quicker. Perhaps re-formatting is a job for an AI programmer. There are just too many different things that can be fed into a program.

So, importing text is an editor's biggest task, and Excellence! doesn't even mention the subject ONCE in its manual. Reformatting also doesn't get much of a mention. Having 1.3 printer drivers and laser-printer capabilities is fine if you can use them, but I output to a daisy-wheel printer, and I haven't seen an Amiga program yet that can do that as well as WordStar on an IBM, which is what I use now. In fact, I have yet to see an Amiga word processor that is even as fast at the job I have to do, let alone faster than WordStar.

I don't intend to bludgeon a newsletter out of an Amiga program unsuited for the job just so we can say that we make our newsletter on an Amiga. After all, the creation and reformatting of text is only a small part of a newsletter - what about the offset printing, folding, inserting in envelopes, bundling up for posting, etc. None of that is done on the Amiga, quite simply because the Amiga isn't the right tool for the job.

Now, I'm not knocking any particular package, I'm just saying that the right tool to do the job I want isn't available for the Amiga yet, and I'll be blowed if I want to spend more time than I already do on the newsletter, using the wrong tool, just to say that it was done on an Amiga.

No doubt this might anger some Amigans, and some might point out programs like Professional Page and other "DeskTop Publishing" packages. And I'd agree that those type of packages are much better for the job that word processors and text editors. BUT, and this is a big but, you need a PostScript laser printer to get the best out of them. Dot matrix output, even from 24 pin LQ-type printers just doesn't cut it. And who in their right mind would authorise the commitment of almost ALL the club's funds on a suitable laser printer, just so we can say the newsletter was produced on an Amiga?

I love my Amiga as much as the next guy - but sadly, it ain't the right tool for the job I've got to do just yet.

==========

The Three Stooges

by Paul Clark

In the past year, a software company by the name of CinemaWare has been producing many games based on the idea of the operator playing an interactive role in a movie; games like Defender of the Crown, Sinbad, SDI and King of Chicago. Although all the games made by CinemaWare have been based around this idea, the Three Stooges is the first to include factual movie characters.

The story behind the game is that Ma's Orphanage is facing financial ruin and a pest of a debt collector (with a very devious laugh) wants to close down the

Orphanage if the debts aren't paid within the month. The Three Stooges come to the rescue and offer to help the Orphanage to get the money. They do this by performing various jobs.

The actual game play involves a hand randomly moving left and right over a group of eight pictures. The pictures represent various tasks you are able to perform. To make life easier on you and your joystick button, you can select a task which slows down the hand's speed. In this task you play the part of Moe and must hit Larry and Curly as much as possible thus reducing the speed of the hand's movement over the pictures.

You can enter a radio quiz which I've found quite hard because each question is based on the Three Stooges. They range from questions like what year Curly stopped performing as a Stooger, to what year they began their television show. One job I particularly enjoy is in the Hospital. This involves racing up the screen on a flat top trolley collecting medical supplies whilst attempting to avoid patients on crutches and in wheel chairs.

The other jobs include working as waiters serving pies in a restaurant, which seems to end up with pie on the Three Stooges more often than not. A boxing match offers a handy sum of money if you are successful, but in this boxing match you don't have to fight. Curly plays a fiddle as inspiration for Larry whilst he is in the ring fighting, the fiddle breaks and Curly must get to the repair shop and back before the fight is over. Your job is to navigate Curly through the obstacles on the footpath, jumping over dogs and dodging rubbish bins and poles. The graphics on this part of the game are excellent; the screen shows the boxing match at the top of the screen and the footpath and obstacles in the bottom half.

The presentation of the game itself is the best I have seen on any of the CinemaWare releases. The atmosphere of the game is great - from the numbers counting down before the beginning of the game like an old movie, to the silent movie tune. The thing that particularly impressed me was the use of digitized voices, there are many of them. The graphic are superb. The Three Stooges are depicted as cartoon characters and look great. The back drops of cityscape are very well presented and there are some very interesting digitised pictures which are brought to life with colour changing in the background. The game is very playable and provides amusing entertainment.

==========

Fortran and the Amiga

by Bryan Pedersen

Old favourites are hard to part with. So when I decided to tackle some serious Amiga programming, only one language came to mind - Fortran.

While C is the most common language of choice for the Amiga, experienced programmers may still rely on their old favourite. I say experienced because most support and examples for Amiga programming revolve around C. Why not go with the tide? Well, Fortran and some other languages have been around for a long time and C does not have the richness of syntax to match these elder languages. Algorithm implementation is well rehearsed for Fortran and standard texts almost always include coded examples. It is unfortunate that the Amiga operating system software has been developed in a rather language dependent way, this could (should) have been avoided.

The Fortran I use is the Absoft AC/Fortran 77. This is an excellent implementation of the language to the 1976 ANSI standard, with some extensions.

Here, I will concentrate on giving a brief description of the implementation, rather than features of the language. Once again, Fortran is a well described language.

Absoft have provided reasonably simple access to all Amiga libraries, such as Exec, Intuition, etc. A typical sequence to open a dos file, for instance, could be:

```
Filename    = "client.dat"//char(0)
File_handle = amiga(Open,Filename,MODE_OLDFILE)
```

Use of Fortran's Implicit None statement helps in controlling the case sensitive nature of variable names.

The distribution disk has include files to declare most (but not all) symbolic names. (Undeclared ones you have to look up yourself and insert into the include files.)

When using this Fortran for an extensive program, it in essential to have the ROM Kernel reference manuals. This is probably true no matter which language you use.

The compiler is fast. With expanded memory (mine is from Subordinate Systems), I regularly achieve figures of over 5000 lines per minute. It may seem rather confusing, but the resulting modules from the compiler are in fact executable, so the above figures actually include at least a partial link as well. If you have many separate files to compile into the one executable, an extra linking step is required however. This is still reasonably quick, though. As an example, with all object modules (from the compiler) in Ram, linking to a single 260k executable takes around 90 seconds.

Assembler routines are very easily incorporated. The 68000 with its addressing modes fits rather nicely with the Fortran convention of passing subroutine arguments by reference. The manual also states that C subroutines and functions may be called, but I have not tried this.

One additional advantage of the Absoft Fortran is that they do have a 68020/68881 version available which is source code compatible with the 68000 version. Therefore programs written in 68000 Absoft Fortran should move straight onto any Amiga processor upgrades (after re-compiling, of course).

Lastly, I should mention that the product is quite mature, having been around since the Amiga was released, now being up to version 2.3. I have talked to some of the software support people at Absoft in the USA, and found them most helpful. I even ordered

an upgrade by phone, quoting a Visa card for payment and had the goods only nine days later. They even paid the postage.

==========

Galileo V2.0 - A Review

By Peter Ward, Padstow NSW

I first used this program during its early days as version 1.0, and frankly was not all that impressed. The program has no copy protection as such, instead a key word from the manual must be entered to access the program. I am pleased to say however, that version 2.0 has ironed out many of the problems initially encountered, and has made this planetarium type program a joy to use on cloudy nights.

Why on cloudy nights? You may well ask... the problem being, despite the fact that Galileo is probably the best Astronomy program available on any personal computer, it still falls short of being an observing aid suitable for serious planning of a night's observing. An extra data disk is now available, obtaining it will cost you US$11.50 and the relevant postage to infinity software in California. To their credit, despite the fact the disk is not included with the original purchase, it only took ten days for my order to be filled! (Aussie retailers please note.) After having obtained said data disk, which in actual fact is the Yale Bright Star Catalog, you have access to over 9000 stars. Not bad, but still not enough for a chart to be used on a night of serious observing. If you are still not convinced, Sky Atlas 2000.0 has about 43,000 stars with 2,500 deep sky objects and the new Uranometria 2000.0 has 332,556 stars alone!

Despite this, some rather nice features are: faster screen "refresh", better accuracy, many, many more stars compared to the 1600 on version 1.0, a mode allows the user to add additional data, such as a comet, supernova or whatever to the data base. Point and center... which means that should you have an interest in a particular patch of sky, simply point the mouse, and the screen is updated around the new point, IDENTIFY! an nice feature which allows you to put the cross hairs over the object of interest and have it "identified" (perhaps the most obvious shortfall of the original), multiple menu selection and a "landscape" viewing mode, which gives a starry sky with a suitable synthetic horizon superimposed, probably being the most visually impressive mode available. Whilst on the various viewing modes, the database of 9,000 stars seems only to be accessed while in "planetarium" mode, which is a shame as other modes serve better the visual observer, when trying to reference a star against the horizon or some known bearing. On the "save to IFF" option, I was a little surprised that the programmer did not simply opt for a straight-out dump of the screen, until I tried this, via other means. Not a good idea with a black sky and tiny white stars... perhaps version 3.0 may tend to this problem of printing star charts. The option left at the moment is to save the screen as per IFF, then resort to a paint program (does anybody not have a copy of DPaint II?) to touch up the original, and make it suitable for use as a star chart.

Another very nice feature allows you to "look down" on the solar system on a particular date. This gives the planets with their respective positions and orbits for that date, with problems of scale being sorted out by having the option of looking at the inner, middle or outer solar system from menu selections. Also hinted at within the instruction manual is a proposed additional data disk which will give some nice graphic pictures of various "deep-sky" objects, and may be accessed via the identify feature. The screen will also update the sky view at from zero to one hundred times the real time rate allowing you to literally watch the night sky roll by, however, all other functions of the program are held up during updates, and this does become tiresome.

Where Version 2.0 really shines is that you could consider it to be a "front end" to the "Yale Bright Star Catalog". With this, you can obtain a data base for variables and B-V colour (though this is spelt color in the U.S. Manual) for the star selected, the distance of the star in light years, and name in terms of "Alpha, Beta etc." indicating relative brightness of the star for the constellation it is in. Spectral class and whether the star is a known variable is also listed. So, should you have an interest in the heavens and live in Melbourne (sorry guys, but my job has moved me to Sydney, the roads may be terrible, but the skies are invariably clear) buy a copy of Galileo, you won't be disappointed.

==========

Amiga Structures - the Heart of the Machine

by Allan Duncan

Just about any code written for the Amiga has to tangle with the veritable jungle of include files and their multiplicity of structure definitions. Since most of the development of the (documented) operating system has been written in the C language, the formal specifications for all the interfaces to the external (ie. your) program environment are given in that language. Translation into assembler is also provided, but since there are but a few souls beavering on that front, and the syntax is obscure, I will leave them to their own devices. For those who don't know, the undocumented operating system is that which we have all come to loathe, AmigaDog, the dreaded BCPL code from MetaComCo. For those of you who thought that the Bantam Books "AmigaDos" book was the one for mere users, and that there was another, more powerful, one for the developers, guess again. That is IT. No further discussion. Finito. Silence.

Enough of this lamenting and on with this discussion on structures. Often a call to the operating system must convey a substantial amount of detail for the required function to be performed. For example, to open a new screen the system wants to know where it is to be placed, how big it is, what pens are to be used for rendering the various borders and gadgets, what sort of screen it is going to be, what font is to be used, what its default title is, what gadgets are to be used, and, finally, a bit-map may be needed for custom screens. Quite a list! Would you like to pass all these through as individual parameters? Time for a quick structure that holds all these bits and pieces, and what better way to pass it in a call than a pointer to same. Since it is necessary for the programmers at C= to all agree on the exact order and type of all the parameters that make up the structure, a header that contains this structure is created and passed around to all that need it, including that special breed, The Developer. Of course, for a suitable fee, CATS (Commodore Amiga Technical Support) will supply it to anyone else who requests this vital information, even though as you read it you find

"CONFIDENTIAL and PROPRIETARY"

on some of the entries. And it might be better to send direct to C= West Chester so that you get the latest, because my seven disk V1.2 set purchased through a local dealer turned out to be a gamma version, supplied in good faith as V1.2.

[Editor's Note: We have just found out that Commodore Australia has instructed Commodore US to refer all direct enquiries from Australia **back** to Australia. So, you **can't** go direct to the US.]

Not only do these include files contain the definitions of structures used in the calls and the resulting returned values, they also contain convenient symbolic names for the multifarious nasty binary constants that are needed to flag this condition or that, as well as what is often considered to be even more important, comments. All of this adds up to a huge pile of documentation on the Amiga - my printout is about one inch thick (and Peter, don't convert that to 25mm, I don't mean more than 24.5mm and less than 26.5mm). [Editor's note: I don't do things like that....] Obviously this provides some difficulty in finding the particular structure that you need at the moment - the intuition.h header is about 22 pages long, and is somewhere in the middle of the stack, and IS the one I am looking for in it anyway? To help those of you who need to delve do so with greater speed, and for the curious, here is a list of all the include files with the structures that they contain (if any). By the time you read this, there will be a copy on the AmigaLink for downloading so that you can machine search it if you feel inclined. Sometimes I would like the entire set joined together so that I could do a search for all the references as well, but I'm lazy, and anyway the file would be so big that I wouldn't have the patience for the long search to complete. As well as the C= supplied includes, I have shown the Manx ones. Enjoy.

clib/macros.h

devices/audio.h
struct IOAudio

devices/bootblock.h
struct BootBlock

devices/clipboard.h
struct ClipboardUnitPartial
struct IOClipReq
struct SatisfyMsg

devices/console.h

devices/conunit.h
struct ConUnit

devices/gameport.h
struct GamePortTrigger

devices/prtgfx.h NEW in 1.3
union colorEntry
struct PrtInfo

devices/input.h

devices/inputevent.h
struct InputEvent

devices/keyboard.h

devices/keymap.h
struct KeyMap

devices/narrator.h
struct narrator_rb
struct mouth_rb

devices/parallel.h
struct IOPArray
struct IOExtPar

devices/printer.h
struct IOPrtCmdReq
struct IODRPReq

devices/prtbase.h
struct DeviceData
struct PrinterData
struct PrinterExtendedData
struct PrinterSegment

devices/serial.h
struct IOTArray
struct IOExtSer

devices/timer.h
struct timeval
struct timerequest

devices/trackdisk.h
struct IOExtTD

exec/alerts.h

exec/devices.h
struct Device
struct Unit

exec/errors.h

exec/exec.h

exec/execbase.h
struct ExecBase

exec/execname.h

exec/interrupts.h
struct Interrupt
struct IntVector
struct SoftIntList

exec/io.h
struct IORequest
struct IOStdReq

exec/libraries.h
extern struct Library

exec/lists.h
struct List

exec/memory.h
struct MemChunk
struct MemHeader
struct MemEntry
struct MemList

exec/nodes.h
struct Node

exec/ports.h
struct MsgPort
struct Message
struct Semaphore

exec/resident.h
struct Resident

exec/semaphores.h
struct Semaphore
struct SemaphoreRequest
struct SignalSemaphore

exec/tasks.h
extern struct Task

exec/types.h

graphics/clip.h
struct Layer
struct ClipRect

graphics/collide.h

graphic/copper.h
struct CopIns
struct cprlist
struct CopList
struct UCopList
struct copinit

graphic/display.h

graphics/gels.h
struct VSprite
struct Bob
struct AnimComp
struct AnimOb
struct DBufPacket
struct collTable

graphics/gfx.h
struct Rectangle
struct BitMap

graphics/gfxbase.h
struct GfxBase

graphics/gfxmacros.h

graphics/graphint.h
struct Isrvstr

graphics/layers.h
struct Layer_Info

graphics/rastport.h
struct AreaInfo
struct TmpRas
struct GelsInfo
struct RasPort

graphics/regions.h
struct RegionRectangle
struct Region

graphics/sprite.h
struct SimpleSprite

graphics/text.h
struct TextAttr
struct TextFont

graphics/view.h
struct ColorMap
struct ViewPort
struct View
struct RasInfo

hardware/adkbits.h

hardware/blit.h
struct bltnode

hardware/cia.h
struct CIA

hardware/custom.h
struct Custom

hardware/dmabits.h

hardware/intbits.h

intuition/intuition.h
struct Menu
struct MenuItem
struct Requester
struct Gadget
struct PropInfo
struct StringInfo
struct IntuiText
struct Border
struct Image
struct IntuiMessage
struct Window
struct NewWindow
struct Screen Moved out in 1.3
struct NewScreen
struct Preferences Moved out in 1.3
struct Remember

intuition/intuitionbase.h
struct IntuitionBase

intuition/preferences.h NEW in 1.3
struct Preferences

intuition/screen.h NEW in 1.3
struct Screen

lattice/ctype.h

lattice/dec.h

lattice/dos.h
struct FCB
struct MELT

lattice/error.h

lattice/fcntl.h

lattice/ios.h
struct UFB

lattice/limits.h

lattice/math.h
struct exception

lattice/setjump.h
struct JMP_BUF

lattice/stdio.h
struct _iobuf
extern struct _iobuf _iob[_NFILE]

libraries/configregs.h
struct ExpansionRom
struct ExpansionControl
struct DiagArea

libraries/configvars.h
struct ConfigDev
struct CurrentBinding

libraries/diskfont.h
struct FontContents
struct FontContentsHeader
struct DiskFontHeader
struct AvailFonts
struct AvailFontsHeader

libraries/dos.h
struct DateStamp
struct FileInfoBlock
struct InfoData

libraries/dosextens.h
struct Process
struct FileHandle
struct DosPacket
struct StandardPacket
struct DosLibrary
struct RootNode
struct DosInfo
struct CommandLineInterface
struct DeviceList
struct FileLock
struct DevInfo NEW in 1.3

libraries/expansion.h

libraries/expansionbase.h NEW in 1.3
struct ExpansionInt
struct ExpansionBase

libraries/filehandler.h
struct FileSysStartupMsg
struct DeviceNode

libraries/mathffp.h

libraries/translator.h

libraries/romboot_base.h NEW in 1.3
struct RomBootBase
struct BootNode

resources/cia.h

manx/assert.h

manx/ctype.h

manx/errno.h

manx/fcntl.h
extern struct _dev {} *_devtab
/* sysio/_main.c AllocMem's it

manx/functions.h

manx/math.h

manx/obj68k.h
struct module
struct newlib
struct symtab

manx/setjmp.h
typedef long jmp_buf[JBUFSIZE]

manx/sgtty.h
struct sgttyb

manx/signal.h

manx/stat.h
struct stat

manx/stdio.h
typedef struct {} FILE
extern FILE Cbuffs[]
/* defined in stdio/getbuff.c */

manx/time.h
struct tm

resources/disk.h
struct DiscResourceUnit
struct DiscResource

resources/misc.h
struct MiscResource

resources/potgo.h

workbench/icon.h

workbench/startup.h
struct WBStartup
struct WBArg

workbench/workbench.h
struct DrawerData
struct DiskObject
struct FreeList
struct WBObject

Modem Run-down

By Darren King

Modems have become a popular item of computer hardware over the past few years. They provide a convenient way to communicate with other computers either around the corner or on the other side of the world. Modems vary in many ways; baud rates (speed at which a modem sends and receives information), Auto answering, Auto dialing and so on.

In this article, I would like to explain the different features, baud rates and quite a lot more about modems, as well as what you can dial and what is available to you via a modem.

What is a modem?

A modem is basically a transmitter and receiver which works over the normal telephone lines. The word MODEM is an abbreviation for two words: MODulator and DEModulator, hence MODEM. There are two types of modems available, the so-called "dumb" modem and the "smart" modem.

The dumb modem is the older of the two, and it is so named because a person must dial the number required, make sure that the phone on the other end is answering, and then switch over from phone to modem before the modem can start transmitting and receiving information. This does take some practice and at first can be a rather complex task. Dumb modems are cheaper than Smart modems, and are better suited to people who only occasionally want to dial up friends or bulletin boards. (More on bulletin boards later.)

Smart modems are the newer range of modems which have been on the market for a few years. Smart modems are called smart because they automatically connect to the phone line, check for dial tone, dial the number and hang up automatically. These modems can also select the right baud rate by themselves as well. This eliminates all of the messing around needed to get a dumb modem connected. However, smart modems cost a lot more than dumb modems, so careful consideration is needed choosing a modem. More on prices and so on later.

Modem Baud Rates

The way in which a modem communicates with another modem is by high-pitched tones. The common speeds (or baud rate) available at the moment are: 300/300, 1200/75, 1200/1200, 2400/2400. These unmeaningful set of numbers are the speed at which a modem sends and receives information. For example, 1200/75 means that the modem will receive data at a rate of 1200 bits of information a second, and will transmit data at the rate of 75 bits per second. The transmitted speed (75 baud) is very slow, but this standard is very useful for mostly receiving information at a high speed, while transmitting only a very small amount of information back.

2400/2400 baud has only become common relatively recently in consumer-type modems, and is rather fast. However, there can be certain problems at this speed because of the telephone lines themselves. The telephone system, as you know, can sometimes give you a bad connection. Sometimes this can cause problems when you are talking to someone, so you can imagine what happens when you are using a modem with precision tones..... All in all though, 2400/2400 is nice if you can get it, since it is so fast and it makes the transfer of a file or a disk relatively quick (a whole Amiga disk of 880K takes around 40 minutes).

The oldest baud rate is 300/300, and has been around for many years. A few years ago, a modem with only 300/300 was considered okay, but these days it is best avoid these modems, as they are almost obsolete. Most people are upgrading to 1200/1200 or 1200/75, and it is best to at least get a modem with 1200/75. Most people consider 300/300 too slow for today's communications purposes. No new modems available today have only 300/300 anyway - they offer at least 300/300 and 1200/75. These modems are best suited for the person who wishes to only get the feel of what a modem can do.

Who or What Can You Talk To?

Once you have your modem, you have to consider what software you will use with it. Diga! by Aegis is a suitable modem program to use with almost all modems available. You can always use other modem programs from bulletin boards later on.

These programs are often at least as good as the commercial programs. A good example is ACCESS. This program is compatible with most modems, and has all the required functions to support both smart and dumb modems.

Always familiarise yourself with the program you with to use and have a good working knowledge of the program before delving into any serious communications with other computers. Remember, if you only have one phone line in your house, then no one else can use the phone while you are using your modem. This can lead to some aggravation from others who wish to use the phone (it does in my house, anyway!).

Now let's look at what is available.

There are many bulletin boards available in Melbourne. There are the occasional ones that require you to register and pay a fee, but generally you do not have to pay a cent (except for the price of a phone call). These bulletin boards (referred to as "BBS" from now on) require you to give your name, address, etc so that verification can be carried out on you. When you are verified, the person who operates the BBS (called the "sysop" or system operator) is just making sure that you are who you say you are. This is generally all that is required before you can use the board properly.

Other things that may be asked when you dial up a BBS for the first time may be "What computer and Modem are you using?" and maybe a few more questions. Most BBSes then allow you to enter the password you will use when you access the system.

When you want to dial up the BBS again at a later date, you only need to enter your name and then your password. You will then have access to the BBS and do whatever you desire.

Services offered by most Bulletin Boards include copying files and programs (UPLOADING and DOWNLOAD-

ING), leaving messages to other users, and countless other information services which can be dumped to your printer or saved to your disk to be retrieved later. This mode is usually called CAPTURE, and is supported by most terminal programs (Diga! and ACCESS have it, anyway). CAPTURE allows you to record all text transmitted and received during a specific time. This is then saved as a text file on your disk that can be examined by a word processor or text editor at a later date.

File Transfer - Uploading and Downloading

Uploading and downloading are the terms given to transferring files to and from a BBS and your computer. You UPload by sending a file to the BBS, and DOWNload by receiving a file from the BBS.

There is normally quite a lot to download from a BBS. The files are all Public Domain (no copyrighted stuff here!), and is usually free for you to copy. Downloading or uploading a file of around 100K will take about 1.1 hours at 300/300, 15 minutes at 1200/1200 and just under 8 minutes at 2400/2400, so you can see that having more than just 300/300 can save you plenty of time.

When you receive a file, it will most likely be ARCHIVED using ARC or ZOO. These are programs that compress a file or series of files, make them an even number of blocks long, and can store all related files in one archived file. This makes it a lot easier to receive and transmit files, and because they are compressed, they take less time to transfer. Once you have received a file, it must be un-arced to recover the original programs.

The software available to you varies from BBS to BBS, but it is all very useful; paint pictures, latest public domain, other modem terminal software, and game hints and tips, just to name a few.

Uploading is greatly encouraged, as it helps expand the library of programs on the BBS. If you regularly upload, then you can sometimes be rewarded with extra time on the BBS, depending on the sysop. BBSes usually have a time limit per call (usually about 45 minutes per user each day), which can be extended if your statistics show that you have contributed.

If you are going to upload, there are a few basic rules (how you follow them is important to the quality of your uploaded software!):

* Do NOT upload commercial software!

* Try to make programs as bug free as possible.

* Include the documentation! Programs and files are usually useless without instructions on how to use them!

* Include source code if possible.

If everyone followed these rules, then the world might just be a little bit nicer.

Australia seems to have very little home-grown public domain software - don't be afraid to upload that little (or large!) program you have written. Someone else might be looking for exactly what you've written.

Which Modem Is Right For Me?

There are many different modems you can buy. But remember, you'll probably want more than just a 300/300 modem, as they are almost obsolete anyway. Also remember that an auto-answer and auto-dial modem will cost quite a lot more than a manual modem (dumb modem). If you only want to check out the BBS scene, then a simple 300/300, 1200/75 dumb modem will be right for you. For the more serious user, then all of the above bauds, plus 1200/1200 and even 2400/2400 as well as auto-answer, auto-dial, etc will be what you are looking for.

The modem must have an RS-232 serial interface so that you can connect it directly to your Amiga. Most modems these days do, but if in any doubt, ask.

Prices? Well, that varies quite a lot. For a simple 1200/75 and 300/300 dumb modem, you should be able to find something for around $100 to $200. For a modem with 300/300, 1200/75, 1200/1200, with auto-answer, auto-dial, auto disconnect, etc, you'll have to spend about $350 to $500. With 2400/2400 as well, you may have to spend at least $650.

What type of modem do I own? It is a kit modem, not a built-up commercial one, and it supports 300/300 and 1200/75, and it is a dumb modem, so I have to dial and set baud rates manually. If you know a bit about electronics, then a kit modem might be a good way to go. Mine cost only $90 and I assembled it in 4 hours. I have used it now for more than four months without any problems.

Moving soon?

Don't forget to tell us!

Every month, Australia Post returns newsletters to us marked "Left Address", "Not At This Address" or "Return to Sender".

To make sure this doesn't happen to your newsletter, please tell us if you move!

If possible, include a mailing label from a past newsletter or your membership number.

Have Modem - Will Dial

OK, so you now have your modem, and your terminal program, but don't put your money away just yet - I'll guarantee you that you will also have an increase in the phone bill! It's easy to get hooked when you first get a modem, however, if you do not make any ISD or STD calls, you will be relatively safe! There are many BBS here in Melbourne for just the price of a local phone call, catering for almost any taste.

Once you have looked around and dialed up many BBSes, you will probably settle on just a few favourites. If you own an Amiga you are lucky, because there are many BBSes around that cater for Amiga software and Amiga interests.

If you put the phone bill aside for the moment and consider the services available to you then a modem is a great idea. Not only can you get good free software, but by just looking and leaving messages to other users you can probably solve some problems with that program or circuit you have, so if you are need some information on something then leave a message. Someone will answer you.

So there you have it! Modems are definitely the way to go these days if you have a computer. If you'd like to leave a messages to me, then I am a regular on AmigaLink and Amiga Limits. Just leave a message under my name, Darren King. I will answer all messages!

Amiga Related Bulletin Boards

Here's a short list of bulletin boards that cater for Amiga users:

AmigaLink	792-3918	24 hrs
Amiga Limits	725-2895	24 hrs
Down Under	429-5819	24 hrs
Physi-chall	427-0424	24 hrs
Thongheads	419-9256	(Weekdays 6pm-9am, and all weekend)

==========

Interceptor Notes

[From AMY Report issues 1 and 2, Copyright 1988 by Henry Colonna and APEInc. All Rights Reserved. Reprinted by permission.]

This next message comes to us from the StarShip Amiga Roundtable on GEnie. It wasn't addressed to AMY Report specifically but has some great information from Mike M. Thanks to GEnie and Mike!

Category 4, Topic 24
Message 41 Tue Jun 14, 1988
MIKEM [- Mike -] at 18:24 PDT

Apparently, EA has been getting mega-calls from owners of Interceptor (I talked to them recently myself).

Since we're talking strange goings-on in the program, I'll relate some of the recent wierdness I've encountered. First, if you inch along the carriers deck you find that you can extend the plane over the edge (almost the entire length of the plane). Taking a look from outside the plane, you'll resemble some sort of funky carrier hood ornament. Similar to this... if you land at SFO with AF-1 already on the ground, and inch your way into its tail section you can take an outside-above view and see that you are resting on top of its tail. Climbing any further than about halfway over AF-1 will first cause you to sink underneath it, and eventually crash into it's midsection (destroying it, but not denting your plane). Also on the AF-1 mission... if you ignore the MIG's and follow AF-1 to SFO, and land at the airfield with it, you can easily take out the MIG's from the runway. They usually wont fire until you are down (or close to it), and even then their missiles will generally hit alongside you somewhere (an interesting sight, by the way). If you do not shoot at the MIG's and let them fly around a bit, they will eventually land on the water (or close) fairly near the runway. This not only gives you easy targets to shoot at, but also an opportunity to view the MIG's close-up.

Currently, I'm keeping occupied in the program by attempting precision parachuting. Already chuted through the Golden Gate, onto Alcatraz Isle, and into E.A.'s plaza. Attempting to scrape the TransAmerica building right now (very hard), and you can really view some terrific up-close scenery this way.

- Mike

The complaints and confusion surrounding the shadow sub mission continue, as I constantly give people hints on how to finish this mission. There is also some heavy criticism for Interceptor from others. Here's a great message from Drew Lucy of CompuServe's AmigaForum:

#: 131003 S7/Games
21-Jun-88 21:41:52
Sb: Intercept has no clothes
Fm: Drew Lucy 76337,75
To: ALL

A while ago someone left a message here saying that they had heard Interceptor was just a demonstration program, intended to prove that the graphics routines would work. Whether this is true or not, it exactly sums up my impression of the program. Interceptor is an amazing example of 3-D graphics on the Amiga. All who worked on it should be commended for this aspect of the program. However, "Radar Raiders" is also and excellent demo of the Amiga's graphics capabilities and I didn't have to pay over $40 for it. Of course, there's nothing to shoot at in Radar Raiders. On the other hand, it does have analog control via the mouse. I wasn't really expecting Interceptor to be a simulation. I was, however, hoping that it would be a game with some long term play value.

I've "flown" Interceptor for only three hours according to my stats and, I'm already bored. I realized that there was a serious problem when the most interesting mission I could think of was trying to shoot down Air Force One with cannon fire. Has anyone been able to shoot down anything with the cannon?

I feel the discussion here about the lack of analog joystick control has preempted discussion of the program's other major flaws. I can accept lack of realism in the name of playability. It doesn't bother me that you bounce off the ground at MACH 1 with your gear up or that the planes hover at their 40,000+ foot ceiling when pointed straight up. It does bother me that Interceptor is deviod of strategy or variation of play.

I'm looking forward to next year when, hopefully, the Interceptor demo will have been turned into a real game or, at least Falcon will have been ported to the Amiga. In the mean time, I'm going back to playing "F-15 Strike Eagle" on the old Atari 800 until something better comes along.

Bitch, bitch, bitch...
Drew

==========

Matt's Csh or Shell

by Peter Billing

After reading other peoples views on CLI enhancements, I thought I would add my two cents worth.

I too use Matt's Csh. I was interested in the article in the July WorkBench by Quentin Black regarding turning the prompt into the name of the current directory. I changed my own startup to include the modification and it works well.

I chose to use Csh after trying some of the others, because I found it the best for what I wanted to do. Here's why:

1. Thirty two commands are written into the program. Of these 15 have direct DOS commands. These are some of the most frequent used and therefore are there when you want them. The program might use 35k but on checking the 15 DOS commands that correspond to them, they added up to 33k and they are the much smaller ARP ones.

2. Function keys can be defined in your startup sequence. Using some of the other CLI enhancers you still need to use a second program to set up the function keys, or not use them.

3. Aliases can be made. The prompt changing was done with an alias as described in the July 1988 Workbench. eg alias wb loadwb Now by just typing wb <ret> the workbench will be loaded. A more useful one is: alias cls echo ^l This will clear the screen.

4. All Dos commands are still available by typing the first character in upper case.

   eg. echo hello will use the shell while
   Echo hello will use the Dos command.

5. There are a number of commands that can be used to help write batch files. I wish I knew how to use them all.

Below is a list of the commands available with the Dos command in brackets if shell uses a different name. The first 15 are the built in Dos commands.

cat (type)	cd	copy	date	dir
devinfo (info)	echo	else	endif	if
mkdir (makedir)	mv (rename)		run	
rm (delete)	source (execute)			

alias	abortline	dec	foreach	forever
goto	help	history	inc	input
label	mem	ps	pwd	quit
return	set	sleep	strhead	strtail
unalias	unset	version		

I hope somebody else can write an article on how to use the extra commands to write batch files.

One useful thing I have just recently learnt how to use is using the .sh extension. If a script file is written with the .sh extension only the first part of the name need be typed like a normal command and the script will be executed.

I have just started working with 'C' on the Amiga after using Turbo Pascal on a CPM system. With a editor and compiler all built in it was very fast at compiling and fixing errors. With the C system you need a Editor, C-Compiler, Assembler and then a Linker. Typing in all these names becomes a bit of a bore. Enter Csh. I made the following script file to do it for me.

```
#  Compiler script #
C _passed.c
A68k  _passed.ASM
Blink _passed.O small.lib to _passed
#  end #
```

I named this file Compile.sh . When I used it I only need to type the following.

```
>Compile test
```

This will automatically be executed. The word 'test' will be passed to the Assembler with the '.ASM' extension, then on to the Linker with the extension '.O' and finally be made into the program 'test'. The C-Compiler I am playing with does not take command arguments. The word '_passed' is a variable built into Csh for passing information around.

==========

STAR-RAY - A Game Review

by Darren King

Okay all you DEFENDER freaks - this is the ultimate "defender" game for the AMIGA! Star Ray is a very good shoot 'em up game - anyone who likes shoot 'em up games will love this one.

For those of you that don't know what the game Defender is like, it was released years ago on the A***I 2600 games machine. The game scrolls around horizontally in a big wrap-around ground. There are many aliens that try to capture the civilians on the ground and attack you. Your job is to destroy all the waves of aliens and defend your people on the ground without shooting the people (this is harder than it sounds!) As you continue through the waves of aliens and other nasties that try to destroy you, more aliens will appear. It is a very challenging game.

Star Ray is a much improved version of Defender. Its graphics are great, both for the Aliens, and the background scenes which have a 3-D scroll effect - the closer background features scroll faster than the far-away scenery. You can easily get hooked on this game!

Star Ray is divided up into two screens: the main playing screen, and a radar of the whole landscape. This way you can see what is coming and line yourself up before the alien appears on the screen.

This game is definitely worth considering - this is how I'd score it:

Graphics:	9/10
Sound:	7/10
Playability:	8/10
"Hook"ability:	8/10
Overall:	8/10

==========

SMAUG Report

By Neil Rutledge

Well the appeal made in the August Workbench seemed to work - we had a large turn out of SMAUGites. The meeting started with yours truly nervously giving a demo of two PD music samples, WOW and the Miami Vice Theme. Before the MV demo, I briefly showed how to install a audio filter switch in the Amiga 500, which I did because the Miami Vice demo switches the filters on and off three times a second, fine if you own a 1000 but an audible nightmare if you use a 500 or 2000. For those who own a 2000, there is raw and filtered audio available at one of the expansion slots. After rebooting, I gave a totally unrehearsed demonstration of the SCA virus. My memory must be failing, I thought I had checked these disks. Such is life.

After a cold start was implemented, a discussion on where SMAUG is and should head was started. The subject of meetings outside of Monash was raised once more and it was decided to restart this useful forum at the residence of yours truly on the Sunday prior to the September meeting. For a report on what we got up to and details on the next one, come to SMAUG at the next AUG meeting.

After the necessary requests for a SMAUG co-ordinator was met with only two offers of assistance (I guess I'm it by default, but I'm willing to release the reins to anyone who offers), we went on to give a introduction to Larry Spiegal's Music Mouse.

Due to the different levels of interest and music experience at the meeting (it ranged from those who wanted to learn what they can do with music, MIDI and the Amiga all the way to professional level), I decided that we would start the next meeting at the basics - how the Amiga produces sounds and the ins and outs of connecting a MIDI keyboard to the AMIGA. After this, I'll demonstrate Music Studio, a bottom-end music program.

At future meetings, we will progress upwards from Music Studio, Sonix, DMCS, Soundscape (and any other music related soft/hardware that I can rope the owner into demoing), finishing with an off Sunday demonstration of a professional setup.

Further info from and ideas to yours truly.

With our editor always requesting us to "write more bloody articles", I've added the following review of Soundscape's Utilities Disk 1.

Review of SoundScape Utilities 1

Two weeks after making a midnight call to Go Amigo in the USA, reciting my Mastercard number to the salesman, I was informed that a article was waiting for me at the post office and if I paid the import duty (can't win them all) I could take it home. At the end of the month I was charged US$49.00 for Soundscape Utilities 1 and US$41.00 for Ferrari Formula 1.

The Utilities disk comes with a 32 page manual which is easier to follow than the original Soundscape manual. Included on the disk are:

MOUSEBENDER — Converts the X-Y axis mouse movements into MIDI information to allow the bending of sampled voices and adding of aftertouch to tapedeck tracks or real time playing.

Select the axis you want the mouse to respond to.
Select Pitchbend and or Aftertouch and the amount.
Select the MIDI Controller No. if needed.
Select the MIDI channel you want to use.

MAPPER/SPLITTER	Allows you to remap MIDI data. An example, assign a different sampled voice to each octave of a MIDI keyboard.
Load, Save	Add & Kill map lists
Input Data Type	Note on/off, Poly Pressure, Control Change, Program Change, Mono Pressure & Pitch Bend
Output Data Type	Same options as Input but the input data type can be changed into a different output type. ie incoming Poly Pressure outputting as PitchBend.
Input Channel #	MIDI channel 1-16.
Output Channel #	MIDI channel 1-16.
Port In	The Soundscape module you want to receive from. The Mapper/Splitter can receive from any module in the right side of the Soundscape Patch Panel.
Port Out	As above but output. ie Any module in the left side of the Soundscape Patch panel.
Min & Max	The lowest and Highest note you want to assign on your keyboard to a specified output channel.
Shift	The level of offset (transposition) for the selected output channel.
FUEL GAUGE	Simply a visual memory gauge showing available "Chip" and " All"

memory. It is the only module that is not patched into the Soundscape Patch Panel.

CLOCKSCALER — This module allows you to change the clock pulse ratio. ie for every two clock pulses in you get one clock pulse out. Useful if you are using Soundscape with met48 but your external MIDI device uses the standard 24 clock pulses to a quarter note. Just select clock pulses in (in the above example example 2) and the new clock pulses out (1).

SYSTEM X — Allows system exclusive data from a MIDI device to be stored as a diskfile. ie saving a drum pattern from a MIDI drum machine or loading a patch into a MIDI Synth. Note, does not allow for handshaking, the dump must be initiated by the MIDI device.

FRAMECOUNTER — Plots a specified score at a specified tempo against the frames per seconds reference used in motion pictures. This allows you to force the tempo of the track so that musical events can be matched to the visual effect.

CPS - Adjust the clocks per second. The tempo that FrameCounter will playback the score against the 24 frames/second film speed.
Start - How far into the score FrameCounter will start.
Count - How long FrameCounter will run after "start".
Do it - Go.

SMOOTHCLOCKER — A replacement for the original clock in Soundscape. Said to be smoother hence giving more accurate results.

INSTALLMODULES — It is possible to use the modules immediately by running Soundscape and double clicking on the module icon you want to use. A similar icon will then appear in the Patch Panel available for use. The same rules apply to patching to the new modules (ie input module patches to output & vice versa) as to the original modules.

To install the modules (allow Soundscape to boot up with the modules in the Patch Panel), you run InstallModules after running Soundscape and loading the modules into the PatchPanel. InstallModules will request the default path for Soundscape and the volume name of the disk which has the modules you want to install. Only the new modules that are currently in the PatchPanel will be installed. InstallModules will make a batch file in your S directory similar to that below -

```
Assign Default: Soundscape:
Run df0:Mousebender
wait 5
run df0:SmoothClocker
wait 5
```

Also included on the Utilities disk is a Public Domain program called Fast Tracker. The addendum to the instructions describes Fast Tracker as a "Front End" program that simplifies the recording and playing back of songs with Soundscape. Basically Fast Tracker is an interface between Soundscape and the user negating the use of the Patch Panel and reducing the complexity of the program. Control over MIDI Channel No. and Sampled voice is obtained from clearly labeled gadgets.

Recording and playing back of tracks is easy with the use of the Fast Tracker tape deck. There is no clock to worry about, and control of Pitch Wheel and After Touch, etc, has been removed to further simplify operation. When only one track is set to record, that track will by default get its MIDI data from MIDI channel 1 and or the Console Keyboard. Drop down menus allow for Loading and saving SMUS or Soundscape tracks. Sampled voices can also be loaded from the menu.

The lower left of the Fast Tracer screen has gadgets to allow control of beats per bar, quantization level, metronome the tempo of playback in beats per minutes. By pushing the Fast Tracker screen to the back, it possible to access Soundscape directly.

I have had a few problems with Fast Tracker. Contrary to the manual which states that Fast Tracker can be run prior to Soundscape, I've had to start Soundscape first then click on the Fast Tracker icon otherwise the program would GURU. Occasional unexplained GURU's have wiped me out whilst I've been playing or recording tracks. It is also interesting to note that the program is slow in reacting to the gadgets. Mimetics make it quite clear in the manual that because this program is Public Domain they cannot take responsibility for technical support or updates.

The Utilities disk is not copy protected and I have had no problems with any of the commercial modules causing GURU's. The biggest problem I've had with the Utilities disk is a very crowded PatchPanel and Low memory warnings. I will be demoing Utilities 1 when we cover Soundscape at a SMAUG meeting.

==========

Latest Public Domain Disks

What luck - saving the newsletter from being just one or two sheets, Fred Fish has released some more disks!

Fish Disk #147

This disk contains the latest version of MicroGNU-Emacs (MG 2b). Many additions and enhancements since the original works by Dave Conroy.

Fish Disk #148

EFJ — "Escape from Jovi" A machine-code game featuring hi-res scrolling, large playfield, disk-based Hi-Score list, stereo sound, and multiple levels. Use a joystick in port 2 to control the ship. Binary only, shareware ($8).

Fme — Nicely done map editor for the Fire-Power (tm) game. Features interlaced hi-res with intuition interface. See the "Readme.fnf" file for information on making a bootable disk. Includes source.

HandyIcons — Adds a menustrip to the WorkBench window that allows you to run selected Workbench Tools by menu selection. Can be set up to provide custom environments. Current version supports only WorkBench Tools and not Projects. Binary only.

Scrambler — A simple program that will encode or decode a text file into illegible gibberish, which resembles executable code, to evade prying eyes. Version 0.01, Binary only.

Fish Disk #149

AnimalSounds — A sample of digitized animal sounds along with a simple sound player.

DX-VoiceSorter — Written to be used with Jack Deckard's VoiceFiler program (Disk 82). It allows for the sorting of a number of voicefiles stored using that program into a new voicefile of voices made up from various files. Includes source.

Keep — A nice little utility program with an intuition interface for BBS and network junkies who download messages in one large file and then read them off-line. Using only the mouse, you can drive through such files a message at a time, examine each at your leisure and tag those you wish to keep. Version 1.2, binary only, but source available with donation to author.

Less — Like Unix "more", only better, with forward and backward scrolling, searching and positioning by percent of file and line number, etc. Now lets you also print the current file. Very useful! This is Amiga version 1.3, an update to the version on disk number 92. Includes source.

Scheme — To quote the ReadMe file: "Scheme is a statically scoped and properly tail-recursive dialect of the Lisp programming language invented by Guy Lewis Steele Jr. and Gerald Jay Sussman". Binary only.

Fish Disk #150

AirFoil — An update to the Airfoil generator on disk #71. Generates airfoil models as well as their corresponding streamline and pressure distributions. Includes source.

DC10 — An AmigaBasic DC-10 instrument flight simulator. Appears to be quite in-depth with flight-planning and take-off options along with an extensive documentation file. Requires re-building on a separate disk and was successfully done so by following the author's instructions in the ReadMe_First file.

ExecLib — A working example of how to build and use user-defined disk-resident libraries. Of special interest to developers working with Lattice C.

Iconizer — A utility program that saves your current mouse pointer to a small icon. You can restore the pointer just by double-clicking on its icon. Allows for building a whole library of pointers and to use them whenever you want. Binary only.

Pilot — An implementation of the PILOT language for the Amiga, including a demo done for the National Park Service. PILOT is a limited use language for use in educational and computer based instruction programs. Binary only with Beta test kit available from authors.

StealMemBoot — A small utility designed to be a direct replacement for NoFastMem kind of programs. It modifies the boot block of a disk, so when you boot with it, all memory allocations will return only CHIP memory.

Fish Disk #151

GlobeDemo — A graphics demo which displays very smooth transitions of the rotating earth. Features a pop-up menu. Includes source.

Icons — Yet another potpourri of interesting icons to choose from if you need one for your own program.

Pcopy — A small intuition-based disk copier similiar to the resident "DiskCopy" except with write-verify and other user-selectable options. Useful for making multiple copies with reliable data. Requires two disk drives. Includes source.

SCT — A CLI-based utility (SetColorTable) for displaying and/or setting a screen's colors. You can save the colors of a screen to be restored later, or copy one screen's colors to another. Includes source.

SlideShow — Very nicely done slide-show program written in assembly language. Features forward/backward presentation and creative screen wipes. Currently works only with IFF lo-res pictures. Executable only along with some new IFF pictures to have come my way. Shareware ($16).

Surveyor — A little utility that opens a window on the current screen and displays information about the pointer. Allows for absolute or relative measurement between two points on the screen. Very handy for precise positioning of icons and such. Includes source.

Fish Disk #152

Blk — A requester making tool employing various recursive algorithms including a recursive parser. It takes input text files and converts them to C-source for including as requester declarations. Includes source.

RunBack — A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Includes source.

UUCP — This is a version of uucp (Unix to Unix Copy Program) for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source.

Fish Disk #153

Dme — Version 1.30 of Matt's text editor. Dme is a simple WYSIWYG editor designed for programmers. It is not a WYSIWYG word processor in the traditional sense. Features include arbitrary key mapping, fast scrolling, title-line statistics multiple windows, and ability to iconify windows. Update to version on disk number 134, includes source.

HP11 — Emulates an HP11C calculator including the program mode. Features an ON/OFF button that turns the calculator into an icon that will sit and wait until you need it again. Documentation on the features is scarce, perhaps some industrious HP owner could write a small tutorial for the benefit of those that don't own an HP calculator. Binary only.

HPMam — A program to manipulate settings and fonts on HP LaserJet+ printers and compatibles. Includes an Intuition interface and some sample picture files. Version 1.0, binary only, shareware.

Synthemania — An interesting, very small (and very persistent!) musical piece. If you plan on stopping it without using three fingers, you better read the document file first! Binary only.

Fish Disk #154

Ada — An Ada Syntax checker for the amiga. Includes lex and yacc source.

AssemblyDemos — A interesting group of assembly language demos for your visual and aural pleasure. Binary only.

DiskLib — Two utilities for those people who like to split up PD disks into disks of different categories. Includes source.

Guardian — Another virus diagnosing and vaccination program. Recognizes any non-standard bootblock. Includes a small utility program to permanently place the program on a copy of your kickstart disk in place of the seldom (if ever!) used Debug() function. Binary only.

PrintSpool — A print-spooling program. Very useful for printing files in the background. Many command-line options. Version 1.0.0, Includes source.

Utilities — A group of four little utility programs, Cal, Undelete DClock and WhereIs. Binary only, see the ReadMe file for a description of each.

VirusX — An update to the virus-detecting program of the same name on disk number 137. This version also checks for the Byte-Bandit strain. Version 1.21, Includes source.

Virus_Alert! — Yet another anti-virus program with a twist. Once installed a message is displayed just after a warm or cold boot notifying the the user that the disk and memory are virus-free, and forcing a mouse-button press before continuing. Anything writing to the bootblock thereafter will destroy the message and a normal virus- infected boot (???) will take-place. Versions 1.01 and 2.01, Binary only.

Wicon — A "Window Iconifier". Allows you to turn your windows into small icons which can be later recalled. Currently installed with MacWin to give your windows a "rubber-banding" effect. Version 1.14, Includes source.

==========

Editor's Column
(Written September 3rd, 1988)

I'm sad. By this time, I would normally have the newsletter finished. This month, however, I received so few articles that I'm beginning to wonder how worthwhile the whole thing is.

We have over 1000 members, yet only a handful ever bother to contribute. This isn't the way a club runs. In fact, it is the way a club goes down the gurgler. I'm more that a bit tired of saying the same thing at meetings each month - "please write articles".

I can't understand what the problem is each month - I'd say that every one of our members uses his or her Amiga at least as much as I do. So what's so hard about writing a few words now and again about whatever it is that you do with your machine. The answer is that it IS NOT hard - for some reason you just don't want to do it.

And THAT is the thing that makes me sad. A small bunch of people bust a gut and use up most of their spare time and energy to make the Amiga Users Group work. A much larger group of people seem to do nothing at all. Why did they join the club? I often wonder, because a club is the sum total of its members - a club is not as good as whatever you can get out of it, a club is as good as what YOU put into it.

PLEASE put something into the club.

Public Domain Software Order Form

Mail to: Amiga Users Group, PO Box 48, Boronia, 3155, Victoria

Disk Numbers:									

Don't forget to specify collection name, ie Fish, Amigan, Amicus, etc

Disks supplied by Amiga Users Group @ $8 each	$
Disks supplied by member @ $2 each	$
Club Use Only:	Total $
Member's Name:	Membership #:
Address:	
	Postcode:

Newsletter Back Issue Order Form

Mail to: Amiga Users Group, PO Box 48, Boronia, 3155, Victoria

Issue Numbers:									

Be patient, we may have to reprint some issues to fill your request

Number of issues ordered @ $2 each	$
Club Use Only:	Total $
Member's Name:	Membership #:
Address:	
	Postcode:

Application for Membership of The Amiga Users Group Inc

Membership is **$20** per year. Send your cheque to: **Amiga Users Group Inc, PO Box 48, Boronia, 3155**

Surname: _____ Details on this side are optional

First Name: _____ Year of birth: _____ Which model Amiga: _____

Address: _____ Occupation: _____

_____ Postcode: _____ Interests: _____

Phone Number: _____ STD Code: _____

Where did you hear about AUG: _____

_____ Dealer's Name: _____

_____ Dealer's Address: _____

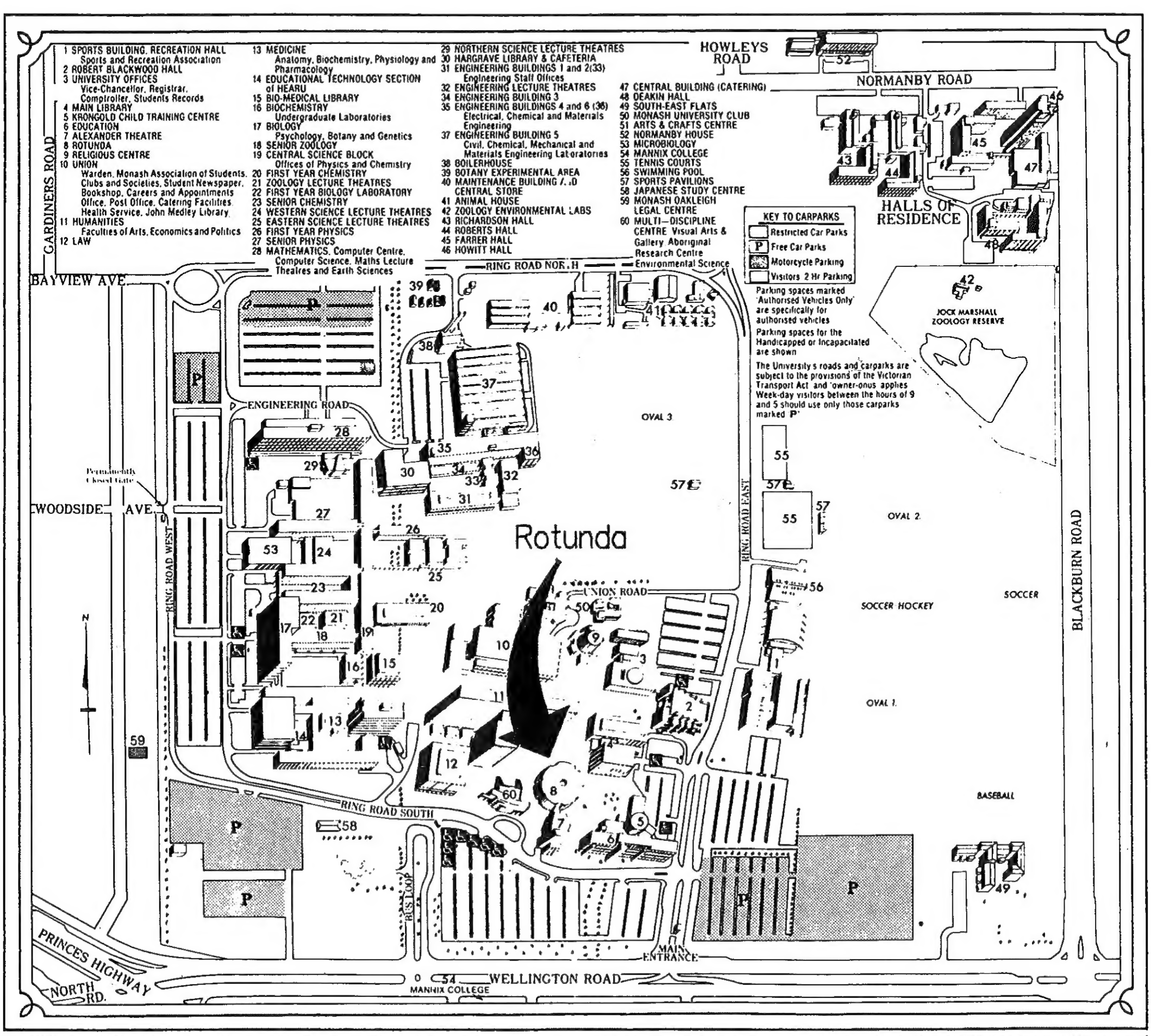
Signed: _____ Date: _____

If admitted as a member, I agree to abide by the rules of the Association for the time being in force.

Club Use Only	Date	Paid	Rcpt #	Memb #	Card Sent

AUG meets on the third Sunday of each month

Monash University is in Wellington Road, Clayton. See Melways Map 70, reference F10. Melways map 84A shows the University Campus in details. I've drawn a huge arrow on the map below to show where the Rotunda is. The best place to park your car is the car park area between Wellington Road and the Rotunda. The entrance to the Rotunda is virtually at the point of the arrow.



BY PUBLIC TRANSPORT . . . The simplest method is to take a train from Flinders Street or Loop stations on the Dandenong/Pakenham line to either Huntingdale or Clayton. Buses run from these stations to the campus or there is a taxi rank at Clayton. With suitable connections the trip takes about 45 minutes — but it can take longer! An inner neighborhood ticket will take you all the way via Huntingdale station and the bus, but you will need to purchase a comprehensive ticket for the trip via Clayton, which encompasses two neighborhoods. The campus is also served by buses from Box Hill, Blackburn, Belgrave, Chadstone, Jells Park-Glen Waverley, Dandenong-Mulgrave, Oakleigh and Elwood.

FROM THE CITY BY CAR . . . An easy route is along St Kilda Road or Kingsway/Queens Road and then on to Dandenong Road. The campus's tall Menzies Building comes into view a kilometre or so before the left turn into Wellington Road on which the main entrance is located. Allow 40-50 minutes for the trip. Drivers should note that restrictions apply in some car parks weekdays 9 a.m. to 5 p.m. and fines **do** apply. There is ample unrestricted parking and, closer to buildings, designated two hour visitor car parks — check the map or ask at the Gatehouse.